UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 09/802,857 | 03/12/2001 | Akihiko Koh | SON-2047 | 3304 |

23353 7590 05/23/2011

RADER FISHMAN & GRAUER PLLC
LION BUILDING
1233 20TH STREET N.W., SUITE 501
WASHINGTON, DC 20036

| EXAMINER | |
|---|---|
| YIGDALL, MICHAEL J | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2192 | |

| MAIL DATE | DELIVERY MODE |
|---|---|
| 05/23/2011 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

Application Number: 09/802,857
Filing Date: March 12, 2001
Appellant(s): KOH ET AL.

Christopher M. Tobin
For Appellant

**EXAMINER'S ANSWER**

This is in response to the appeal brief filed on March 10, 2011 appealing from the Office action mailed on December 16, 2010.

**(1) Real Party in Interest**

The examiner has no comment on the statement, or lack of statement, identifying by name the real party in interest in the brief.

**(2) Related Appeals and Interferences**

The examiner is not aware of any related appeals, interferences, or judicial proceedings, other than those identified in the appellant's brief, that will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

**(3) Status of Claims**

The following is a list of claims that are rejected and pending in the application:

Claims 63-69.

**(4) Status of Amendments after Final**

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

**(5) Summary of Claimed Subject Matter**

The examiner has no comment on the summary of claimed subject matter contained in the brief.

**(6) Grounds of Rejection to Be Reviewed on Appeal**

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the

appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

**(7) Claims Appendix**

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

**(8) Evidence Relied Upon**

| | | |
|---|---|---|
| 5,784,537 | SUZUKI et al. | 7-1998 |
| 5,357,627 | MIYAZAWA et al. | 10-1994 |

**(9) Grounds of Rejection**

The following ground(s) of rejection are applicable to the appealed claims:

- Claim 63 stands rejected under 35 U.S.C. § 103(a) as unpatentable over U.S. Patent No. 5,784,537 to Suzuki et al. ("Suzuki") in view of U.S. Patent No. 5,357,627 to Miyazawa et al. ("Miyazawa").

Claim 63

Suzuki teaches a data processing apparatus comprising:

a central processing unit (see, for example, CPU 14 in FIG. 1) configured to initiate execution of an interrupt processing routine upon a transition of an interrupt request signal (see, for example, step S11 in FIG. 4B and column 6, lines 27-36, which shows that the CPU 14 is

configured to execute interrupt processing upon transition of an interrupt request signal), a first signal and a second signal being input to said central processing unit as said interrupt request signal (see, for example, step S5 in FIG. 4A, steps S28, S29 and S30 in FIG. 4B, and column 6, line 65 to column 7, line 2, which shows that S number of interrupt request signals are input to the CPU 14),

wherein said central processing unit executes a program code, said program code being stored in memory at a program address (see, for example, column 3, lines 64-67, which shows that the CPU 14 executes program code stored in ROM 18, inherently at a program address).

To the extent Suzuki does not explicitly describe that said memory is random access memory, such an implementation nonetheless would have been obvious to those of ordinary skill in the art. For example, Suzuki teaches a random access memory for storing data when the program code is executed (see, for example, RAM 16 in FIG. 1 and column 4, lines 1-5). Moreover, in an analogous art, Miyazawa teaches a correction program RAM 74 that stores correction program data (see, for example, FIG. 14 and column 11, lines 7-13).

A person of ordinary skill in the art could, with predictable results, implement the data processing apparatus of Suzuki such that said memory is a random access memory. Therefore, as Suzuki and Miyazawa suggest, the claimed subject matter would have been obvious to those of ordinary skill in the art at the time the invention was made.

To the extent Suzuki in view of Miyazawa does not explicitly describe that a counter register is located within said random access memory, said counter register being incremented when said program address coincides with said first bug address or said second bug address, such an implementation nonetheless would have been obvious to those of ordinary skill in the art.

For example, Suzuki teaches a value in the two most significant bits of a register that represents a number of parts of a program to be corrected, or a number of correction addresses (see, for example, Suzuki, FIG. 2B and column 4, lines 41-49), and further teaches that such values are stored in the RAM 16 when the program code is executed (see, for example, column 4, lines 1-5). Moreover, Suzuki teaches storing a value that represents the number of correction addresses and counting down the stored value each time the program address coincides with a correction address, such that the value of the count subtracted from S represents the number of times the addresses coincide (see, for example, step S5 in FIG. 4A and step S28 in FIG. 4B).

In Suzuki, the count (i.e., the counter register) is decremented when the program address coincides with a correction address. However, a person of ordinary skill in the art could, with predictable results, implement the data processing apparatus of Suzuki such that the counter register is incremented rather than decremented. Specifically, with reference to FIGS. 4A and 4B, given the number of correction addresses S, a person of ordinary skill in the art could initialize the value to 0 (rather than S) in step S5, increment (rather than decrement) the stored value in step S28, and check whether the stored value is equal to S (rather than 0) in step S29. Such an implementation is analogous to the one described in Suzuki and provides the intended results. Indeed, as evidenced in the Suzuki reference, incrementing the value of a counter register is within the level of ordinary skill in the art (see, for example, step S88 in FIG. 12B and column 9, lines 52-54). In such an implementation, the stored value (i.e., the value of the counter register) would represent the number of times the addresses coincide.

Therefore, as Suzuki suggests, it would have been obvious to those of ordinary skill in the art at the time the invention was made to implement the data processing apparatus of Suzuki

such that a counter register is located within said random access memory, said counter register being incremented when said program address coincides with said first bug address or said second bug address.

- Claims 64-69 stand rejected under 35 U.S.C. § 102(b) as anticipated by U.S. Patent No. 5,784,537 to Suzuki et al. ("Suzuki").

Claim 64

Suzuki teaches a data processing apparatus comprising:

a central processing unit (see, for example, CPU 14 in FIG. 1) configured to initiate execution of an interrupt processing routine upon a transition of an interrupt request signal (see, for example, step S11 in FIG. 4B and column 6, lines 27-36, which shows that the CPU 14 is configured to execute interrupt processing upon transition of an interrupt request signal), a first signal and a second signal being input to said central processing unit as said interrupt request signal (see, for example, step S5 in FIG. 4A, steps S28, S29 and S30 in FIG. 4B, and column 6, line 65 to column 7, line 2, which shows that S number of interrupt request signals are input to the CPU 14),

wherein said central processing unit executes a program code, said program code being stored in memory at a program address (see, for example, column 3, lines 64-67, which shows that the CPU 14 executes program code stored in ROM 18, inherently at a program address),

wherein said first signal indicates when said program address and a first bug address coincide, said second signal indicating when said program address and a second bug address

coincide (see, for example, column 6, lines 27-36, which shows that each of the S number of signals indicates that the program address and a correction address coincide).

Claim 65

The rejection of claim 64 is incorporated, and Suzuki further teaches that a first coincidence detecting circuit compares said program address with said first bug address, said first coincidence detecting circuit outputting said first signal when said program address and said first bug address coincide (see, for example, ROM correction processing circuit 24 in FIG. 1 and column 4, lines 12-26, which shows that the circuit compares the program address with the correction address and outputs the signal when the program address and the correction address coincide).

Claim 66

The rejection of claim 65 is incorporated, and Suzuki further teaches a second coincidence detecting circuit compares said program address with said second bug address, said second coincidence detecting circuit outputting said second signal when said program address and said second bug address coincide (see, for example, ROM correction processing circuit 24 in FIG. 1 and column 4, lines 12-26, which shows that the circuit compares the program address with the correction address and outputs the signal when the program address and the correction address coincide).

Claim 67

The rejection of claim 66 is incorporated, and Suzuki further teaches that a number of times said first and second bug addresses coincide with said program address is counted, a value

representing said number of times (see, for example, FIG. 2B and column 4, lines 41-49, which shows a value in the two most significant bits of a register that represents a number of parts of a program to be corrected, or the number of correction addresses, and see, for example, step S5 in FIG. 4A and step S28 in FIG. 4B, which shows storing the value and counting down the stored value each time the program address coincides with a correction address, such that the value of the count subtracted from S represents the number of times the addresses coincide).

Claim 68

The rejection of claim 67 is incorporated, and Suzuki further teaches or suggests that said first bug address indicates a starting address for a first buggy part of a program or data, said second bug address indicating a starting address for a second buggy part of said program or data (see, for example, column 5, lines 10-14, which shows that each correction address indicates a starting address for a part of a program to be corrected).

Claim 69

The rejection of claim 68 is incorporated, and Suzuki further teaches or suggests that said first buggy part or said second buggy part is selected for correction, said central processing unit using said value to select said first buggy part or said second buggy part (see, for example, column 6, line 65 to column 7, line 7, which shows that the part of the program to be corrected is selected for correction based on the value).

**(10) Response to Arguments**

Appellant argues that Suzuki "fails to disclose, teach, or suggest first and second signals input to the central processing unit as an interrupt request signal" (brief, page 11).

However, the examiner does not agree with Appellant's conclusion. Suzuki clearly teaches an interruption (interrupt) request signal 34 input to the central processing unit 14 (see, e.g., FIG. 1 and column 4, lines 16-18). Here, Appellant contends that Suzuki "fails to disclose, teach, or suggest the interruption request signal 34 being first and second signals" (brief, page 12). However, in Suzuki, the interruption request signal 34 is generated a plurality of times. Suzuki describes that the interruption request signal 34 is generated S number of times for the S number of correcting portions in a program module (see, e.g., step S5 in FIG. 4A; steps S28, S29 and S30 in FIG. 4B; column 6, lines 27-36 and column 6, line 65 to column 7, line 2). In other words, S number of interruption request signals 34 are input to the CPU 14.

For example, Suzuki illustrates "an operation for correcting two portions" of a program module (see, e.g., FIGS. 12A and 12B, and column 9, lines 17-25; emphasis added). Suzuki clearly describes the generation of a "first correction execution interruption" and the generation of a "second correction execution interruption" (see, e.g., column 9, lines 44-51 and column 9, line 66 to column 10, line 4; emphasis added). The first and second interruption request signals are input to the CPU 14 as the interruption request signal 34.

Appellant is respectfully reminded that the claims are to be given the broadest reasonable interpretation consistent with the specification. See MPEP § 2111. As pointed out in the Office action, the language of the claims does not limit them to the embodiment shown in Figure 7 (provided in Appellant's brief at page 10). The claims do not specify that the "first signal" and the "second signal" necessarily represent separate signal paths, and do not particularly define how the signals are input to the central processing unit "as" the interrupt request signal. A broad and reasonable interpretation of claim 64, for example, is that the "first" and "second" signals

represent first and second instances of the same signal, or represent the same signal at first and second points in time. Indeed, at page 5 of the decision mailed on May 14, 2010, the Board of Patent Appeals and Interferences states:

> Lastly, we turn to independent claim 45. This claim has several recitations of first and second as applied to interrupt request signals, coincidence detecting circuits, bug addresses, and the like among several interrelated clauses. The claim fails to specify that the first and second recitations in the various clauses are with respect to different entities or elements. The first and second elements may apply to a single given circuit element at different points in time.

Thus, the examiner submits that Suzuki anticipates "a first signal and a second signal being input to said central processing unit as said interrupt request signal" such as recited in claim 64.

Appellant argues that Suzuki "fails to disclose, teach, or suggest a second coincidence detecting circuit" (brief, page 13).

However, the examiner does not agree with Appellant's conclusion. Suzuki clearly teaches a coincidence detecting circuit in the form of ROM correction processing circuit 24 (see, e.g., FIG. 1 and column 4, lines 12-26). Here, Appellant contends that Suzuki "fails to depict the ROM correction processing circuit 24 having more than one PC comparison register section 20" and that instead, "only a single PC comparison register section 20 is disclosed" (brief, page 13). However, in Suzuki, the ROM correction processing circuit 24 also includes a PC value latch section 22. To generate the S number of interruption request signals 34 noted above, Suzuki further describes that the PC value latch section 22 is updated S number of times for the S number of correcting portions in the program module (see, e.g., step S6 in FIG. 4A; steps S28, S29 and S30 in FIG. 4B; column 6, lines 6-11 and column 7, lines 2-7). In other words, Suzuki teaches S number of instances of the ROM correction processing circuit 24.

As pointed out above, the claims are to be given the broadest reasonable interpretation consistent with the specification. See MPEP § 2111. The language of the claims does not limit them to the embodiment shown in Figure 7 of Appellant's specification. Thus, the examiner submits that Suzuki anticipates "a first coincidence detecting circuit" and "a second coincidence detecting circuit" such as recited in claims 65 and 66.

Appellant argues that Suzuki "fails to disclose, teach, or suggest a number of times the first or second bug address has coincide[d] with the program address being counted, a value representing the number of times" (brief, page 14).

However, the examiner respectfully submits that Appellant's analysis does not address the reasoning actually set forth in the Office action. In Suzuki, Figure 2B illustrates two bits that are set to a value representing the number of correcting portions in a program module (see, e.g., column 4, lines 41-49). Suzuki illustrates that the number of correcting portions, S, is stored in step S5 (see, e.g., FIG. 4A). The ROM correction interruption processing (routine) is executed every time the program counter address coincides with the correction execution interruption address (i.e., the "bug address") of a correcting portion (see, e.g., column 4, lines 12-26 and column 6, lines 27-36). Then, the stored number of correcting portions S is decremented (see, e.g., step S28 in FIG. 4B and column 6, lines 61-62). In other words, Suzuki teaches "counting down" the stored number of correcting portions S every time the addresses coincide. Thus, Suzuki anticipates the limitation that "a number of times said first and second bug addresses coincide with said program address is counted" such as recited in claim 67.

The claim further recites "a value representing said number of times." Contrary to Appellant's implication, the examiner does not suggest that the "number of use bytes" described

in Suzuki represents the number of times the addresses coincided (brief, page 16). Likewise, the examiner does not suggest that the number of correcting portions S represents the number of times the addresses coincided (brief, page 17). Instead, as set forth in the Office action, the stored number of correcting portions S subtracted from the initial number of correcting portions S represents the number of times the addresses coincided. Thus, Suzuki anticipates "a value representing said number of times" such as recited in the claim.

For example, consider Figure 3 of Suzuki. If there are "three portions to be corrected" in the program module (see, e.g., column 5, lines 10-11), then the initial number of correcting portions S stored in step S5 would be 3. After executing the ROM correction interruption processing one time, the stored number of correcting portions S would be decremented, in step S28, to 2. Therefore, the stored number of correcting portions S subtracted from the initial number of correcting portions S would be 3 – 2 = 1, representing the number of times the addresses coincided (i.e., one time).

Appellant argues that Suzuki "fails to disclose, teach, or suggest the central processing unit using the value to select the first buggy part or the second buggy part" (brief, page 18).

However, the examiner does not agree with Appellant's conclusion. In Suzuki, the correcting portions of a program module represent the "buggy parts" of the program module. Suzuki describes that the CPU 14 executes a correction program to correct a selected one of the correcting portions (see, e.g., step S26 in FIG. 4B and column 6, lines 44-57). The correcting portion is selected based on the correction execution interruption address stored in the PC value latch section 22 of the ROM correction processing circuit 24 (see, e.g., column 6, lines 6-11 and 27-36). The correction execution interruption address is stored in the PC value latch section 22

based on the "value" noted above (i.e., the value representing the number of times the addresses coincided). Specifically, Suzuki describes:

> Then, the stored number of correcting portions S is decremented (step S28). As a result of the decrement, it is checked whether or not the number of correcting portions is 0. (Column 6, lines 61-64.)
>
> If the number of correcting portions S is 0, the processing goes to step S33 since there is no residual correcting portion in the subroutine module of code No. m. In step S29, if the number of correcting portions S is not 0, the processing goes to step S30 since there is still residual correcting portions. If the correction program of step S26 is nth—first correcting portion, correction interruption address of the nth correcting portion (next correcting portion) is read from the EEPROM 28 and stored in the PC value latch section 22 of the ROM correction processing circuit 24 (step S30). (Column 6, line 65 to column 7, line 7.)

Thus, the examiner submits that Suzuki anticipates the limitation "wherein said first buggy part or said second buggy part is selected for correction, said central processing unit using said value to select said first buggy part or said second buggy part" such as recited in claim 69.

Appellant argues that the combination of Suzuki and Miyazawa "fails to disclose, teach, or suggest a counter being incremented when the program address coincides with a first bug address or a second bug address" (brief, page 23).

However, the examiner does not agree with Appellant's conclusion. Appellant acknowledges that Suzuki describes that the stored number of correcting portions S is decremented in step S28 (brief, page 25). In other words, Suzuki describes decrementing or counting down a counter or counter register. The stored number of correcting portions S (i.e., the counter register) is decremented in step S28 as part of the "ROM correction data setting processing for [the] next correcting portion" (see, e.g., column 6, lines 58-62), as a result of executing the ROM correction interruption processing (see, e.g., FIG. 4B). As noted above, the

ROM correction interruption processing is executed every time the program counter address coincides with the "bug address" of a correcting portion (see, e.g., column 4, lines 12-26 and column 6, lines 27-36). Thus, Suzuki teaches or suggests decrementing a counter register when the program counter address coincides with a bug address.

The examiner appreciates that claim 63 recites "said counter register being incremented" rather than decremented. Nonetheless, as reasoned in the Office action, a person of ordinary skill in the art could, with predictable results, implement the data processing apparatus of Suzuki such that the counter register is incremented (i.e., counted up) rather than decremented (i.e., counted down). Specifically, with reference to Figures 4A and 4B of Suzuki, given the number of correcting portions S, a person of ordinary skill in the art could set the initial value to 0 (rather than S) in step S5, increment (rather than decrement) the stored value in step S28, and check whether the stored value is equal to S (rather than 0) in step S29. Such an implementation is analogous to the one described in Suzuki and provides the intended results. Indeed, as evidenced in the Suzuki reference, incrementing the value of a counter register is within the level of ordinary skill in the art (see, e.g., step S88 in FIG. 12B and column 9, lines 52-54). In such an implementation, the stored value (i.e., the value of the counter register) would represent the number of times the addresses coincided.

As Appellant notes (brief, page 26), a person of ordinary skill has good reason to pursue the known options within his or her technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. See *KSR Int'l Co. v. Teleflex Inc.*, 82 USPQ2d 1385, 1397 (U.S. 2007). Here, Appellant contends that "no objective evidence has been identified … for showing that incrementing a counter when the

program address coincides with a first bug address or a second bug address would have been know[n]" (brief, page 26), and similarly contends that "no objective evidence has been identified … for showing that a person of ordinary skill in the art could, with predictable results, implement the data processing apparatus of Suzuki such that the counter register is incremented rather than decremented" (brief, page 26).

However, the teachings of Suzuki are, in fact, evidence that incrementing a counter register would have been known to those of ordinary skill in the art and would have yielded predictable results. As set forth in the Office action, Suzuki clearly demonstrates that incrementing a counter is within the level of ordinary skill (see, e.g., step S88 in FIG. 12B and column 9, lines 52-54). Indeed, Appellant concedes that Suzuki describes that a counter is incremented in step S88 (brief, page 24). The examiner points out that incrementing and decrementing are analogous operations (i.e., counting up versus counting down). Thus, contrary to Appellant's allegation (brief, page 27), the reasoning supporting the conclusion of obviousness is based on ordinary skill and common sense, and does not improperly rely on hindsight.

The examiner respectfully submits that Appellant's citation to *Ex parte Givens* is misplaced (brief, page 26). The decision in *Ex parte Givens* and the explanation that "a summer is an additive circuit and not a subtractive circuit" relate to adaptive filtering algorithms, not to incrementing and decrementing operations. Moreover, the decision in *Ex parte Givens* relates to anticipation and not to obviousness.

**(11) Related Proceeding(s) Appendix**

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Michael J. Yigdall/
Primary Examiner
Art Unit 2192

Conferees:

/Tuan Q. Dam/
Tuan Q. Dam
Supervisory Patent Examiner, Art Unit 2192

/Lewis A. Bullock, Jr./
Supervisory Patent Examiner, Art Unit 2193